

## Задача А. Безумие и Отвага

Имя входного файла: `heroes.in`  
Имя выходного файла: `heroes.out`  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 512 мегабайт

Многие из нас с детства мечтали создавать компьютерные игры, а для некоторых это даже стало причиной, по которой они начали изучать информатику и программирование. Мишина мечта сбылась, и теперь он работает в известной и уважаемой корпорации «Метель», выпустившей в своё время такие шедевры, как «Искусство войны» и «Звёздное ремесло».

Недавно Миша присоединился к проекту новой ролевой игры «Безумие и отвага». Её ключевой особенностью является возможность на каждом из уровней заново выбирать персонажа для его прохождения.

Перед стартом очередного уровня игроку доступны  $N$  героев. Каждый герой характеризуется силой атаки  $a_i$  и запасом здоровья  $b_i$ . Уровень представляет собой длинную пещеру, содержащую  $M$  монстров. Каждый монстр также имеет свою силу атаки  $c_i$  и запас здоровья  $d_i$ . Зайдя в пещеру, герой сначала сражается с первым монстром, затем, если остаётся жив, сражается со вторым и так далее, пока не погибнет или не дойдёт до конца. Количество жизней героя не восстанавливается между боями, то есть каждую следующую драку он начинает с меньшим запасом здоровья, чем предыдущую.

Бой между монстром и героем состоит в одновременном обмене ударами. Каждый из них, нанося удар, уменьшает запас здоровья противника на величину, равную силе своей атаки. Как только запас здоровья кого-либо из сражающихся становится неположительным, он умирает, и бой прекращается. Обратите внимание, что при такой схеме боя возможна ситуация, когда оба противника погибнут одновременно.

Компания планирует распространять игру бесплатно, получая доход за счёт продажи разнообразных бонусов, реализовать один из которых и поручено Мише. Данный бонус позволяет игроку узнать, сколько монстров убьёт каждый из героев, если игрок выберет именно его для прохождения данного уровня. Так как монстров и героев может быть очень много, Миша столкнулся со сложностями при вычислении необходимых значений и обратился за помощью к вам.

### Формат входных данных

В первой строке ввода записаны два целых числа  $N$  и  $M$  — количество доступных игроку героев и количество монстров в пещере соответственно ( $1 \leq N, M \leq 200\,000$ ).

Следующие  $N$  строк описывают героев. Каждая из них содержит два целых числа  $a_i$  и  $b_i$ , задающих силу атаки и запас здоровья  $i$ -го героя ( $1 \leq a_i, b_i \leq 10^9$ ).

Далее следуют  $M$  строк, описывающих находящихся в пещере монстров. Каждое описание состоит из двух целых чисел  $c_i$  и  $d_i$ , обозначающих параметры  $i$ -го монстра ( $1 \leq c_i, d_i \leq 200\,000$ ). Порядок расположения монстров в пещере совпадает с порядком их описания, то есть первым необходимо убить монстра, описанного в строке  $N + 2$ , а последним — в строке  $N + M + 1$ .

### Формат выходных данных

Выведите  $N$  чисел по одному в строке.  $i$ -я строка должна содержать ответ для  $i$ -го героя.

### Примеры

heroes.in	heroes.out
5 3	0
1 2	1
2 2	2
10 10	3
100 10	3
1 100	
2 2	
7 2	
3 20	

## Замечание

Бой между первым героем и первым монстром в пещере продлится один ход, после которого герой погибнет, а монстр останется в живых.

Параметры второго героя совпадают с параметрами первого монстра, поэтому они убьют друг друга на первом же ходу боя. Ответ для данного героя равен одному.

Если игрок выберет для прохождения уровня третьего героя, то после боя с первым монстром его запас здоровья будет равен восьми, а после боя со вторым — единице. Для убийства третьего монстра ему необходимо сделать два удара, но он умрёт после первой же его атаки.

У четвёртого героя столько же жизней, сколько и у третьего, но сила атаки гораздо больше, поэтому он пройдёт уровень полностью, хотя и погибнет в последней драке.

Пятый герой обладает минимально возможной силой атаки, но при этом у него большой запас здоровья, поэтому он сможет пройти весь уровень и остаться в живых. После первого боя его запас здоровья будет равен 96, после второго — 82, а в конце игры останется только 22.

## Задача В. Переселение

Имя входного файла: `blocks.in`  
Имя выходного файла: `blocks.out`  
Ограничение по времени: 10 секунд  
Ограничение по памяти: 256 мегабайт

В общежитии, где разместили преподавателей летних сборов по чёрной магии и колдовству, проводится ремонт. Поэтому время от времени людям, проживающим в какой-то комнате, приходится переселяться в какую-то другую комнату.

Для переселения разработан план — список троек чисел  $(T_i, A_i, B_i)$ : вечером дня  $T_i$  все жители комнаты  $A_i$  переселяются в комнату  $B_i$ . Жители всех остальных комнат, в том числе  $B_i$ , при этом остаются на своих местах.

Но не всё так просто: иногда в план вносятся изменения. Каждое изменение — это добавление тройки  $(T_i, A_i, B_i)$ . Известно, что до и после каждого изменения все  $T_i$ , присутствующие в плане, различны.

К администрации общежития иногда поступают вопросы: в какой комнате будет утром дня  $R_i$  по плану жить человек, если он заселяется утром дня  $L_i$  в комнату  $S_i$ . Реализуйте программу, которая отвечает на такие вопросы.

### Формат входных данных

В первой строке задано два числа  $n$  и  $m$  ( $1 \leq n, m \leq 10^5$ ) — количество комнат и начальное количество перемещений. Следующие  $m$  строк содержат по три числа  $T_i, A_i, B_i$ .

Следующая строка задает число  $k$  ( $1 \leq k \leq 10^5$ ) — количество запросов. Следующие  $k$  строк задают запросы одного из двух видов

- `?  $t_1 t_2 a$`  — сказать, где будет находиться утром дня номер  $t_2$  человек, заселившийся утром дня номер  $t_1$  в комнату  $a$ .
- `+  $tab$`  — добавить в план переселения операцию перемещения вечером дня номер  $t$  всех жильцов комнаты  $a$  в комнату  $b$ .

Комнаты нумеруются от 0 до  $n - 1$ , индексы массивов от 0. Номера дней — целые числа от 0 до  $10^9$ .

### Формат выходных данных

Для каждого запроса выведите одно число — ответ на задачу.

### Примеры

<code>blocks.in</code>	<code>blocks.out</code>
3 3	1
10 0 1	1
20 0 2	0
30 2 1	2
11	2
? 10 20 0	0
? 10 21 0	2
? 11 20 0	0
? 11 21 0	0
? 20 21 0	1
+ 25 2 0	
? 21 25 0	
? 20 25 0	
? 20 26 0	
? 20 26 2	
? 0 1000000000 1	

## Задача С. Мощный массив

Имя входного файла: `power.in`  
Имя выходного файла: `power.out`  
Ограничение по времени: 5 секунд  
Ограничение по памяти: 256 мегабайт

Имеется массив натуральных чисел  $a_1, a_2, \dots, a_n$ . Рассмотрим некоторый его подмассив  $a_l, a_{l+1}, \dots, a_r$ , где  $1 \leq l \leq r \leq n$ , и для каждого натурального числа  $s$  обозначим через  $K_s$  число вхождений числа  $s$  в этот подмассив. Назовем *мощностью* подмассива сумму произведений  $K_s \cdot K_s \cdot s$  по всем различным натуральным  $s$ . Так как количество различных чисел в массиве конечно, сумма содержит лишь конечное число ненулевых слагаемых.

Необходимо вычислить мощности каждого из  $t$  заданных подмассивов.

### Формат входных данных

Первая строка содержит два целых числа  $n$  и  $t$  ( $1 \leq n, t \leq 200000$ ) — длина массива и количество запросов соответственно.

Вторая строка содержит  $n$  натуральных чисел  $a_i$  ( $1 \leq a_i \leq 10^6$ ) — элементы массива.

Следующие  $t$  строк содержат по два натуральных числа  $l$  и  $r$  ( $1 \leq l \leq r \leq n$ ) — индексы левого и правого концов соответствующего подмассива.

### Формат выходных данных

Выведите  $t$  строк, где  $i$ -ая строка содержит единственное натуральное число — мощность подмассива  $i$ -го запроса.

### Примеры

power.in	power.out
3 2	3
1 2 1	6
1 2	
1 3	
8 3	20
1 1 2 2 1 3 1 1	20
2 7	20
1 6	
2 7	

## Задача D. Важные гости

Имя входного файла: `guests.in`  
Имя выходного файла: `guests.out`  
Ограничение по времени: 6 секунд  
Ограничение по памяти: 512 мегабайт

Вероятно, вы знакомы с известным танцором, мастером Падеграсом, готовящим сейчас студентов Мводского Государственного Университета к выпускному балу. Завтра к главе университета придут важные зарубежные гости, желающие посмотреть на местную студенческую жизнь. Поэтому, Падеграса попросили выбрать небольшое подмножество студнентов, чтобы они продемонстрировали, как умеют танцевать.

В группе профессора учатся  $N$  мальчиков и  $M$  девочек, кроме того, он знает все  $K$  пар взаимных симпатий между ними. Для запланированного танца Падеграсу нужно выбрать ровно двух мальчиков и ровно двух девочек. Так как танец будет очень страстным и энергичным, танцевать может только пара, обладающая взаимной симпатией. Кроме того, во время танца пары будут много раз меняться, необходимо, чтобы каждая из четырёх возможных пар из мальчиков и девочек обладала взаимной симпатией.

Ещё одной проблемой является то, что встреча произойдёт уже завтра, и Падеграс не уверен, кто из его учеников сможет прийти. Подбодрите его и посчитайте, сколько существующих различных множеств из ровно двух мальчиков и ровно двух девочек, удовлетворяющих всем требованиям.

### Формат входных данных

В первой строке содержатся целые числа  $N$ ,  $M$  и  $K$ : количество мальчиков, количество девочек и количество взаимных симпатий между ними ( $1 \leq N, M \leq 100\,000$ ,  $1 \leq K \leq 200\,000$ ).

В каждой из следующих  $K$  строк содержатся по два числа, описывающих симпатию:  $v_i$  ( $1 \leq v_i \leq N$ ) и  $u_i$  ( $1 \leq u_i \leq M$ ) означают, что между мальчиком  $v_i$  и девочкой  $u_i$  есть симпатия.

### Формат выходных данных

Выведите единственное число: ответ на задачу

### Примеры

<code>guests.in</code>	<code>guests.out</code>
3 4 9 1 1 1 2 1 4 2 1 2 3 3 1 3 2 3 3 3 4	4